بسم الله الرحمن الرحيم

بقلم / أحمد مناع

درس اليوم إن شاء الله عن الكائن ..... أهلا وسهلا بكم إخوانى الكرام Threads .... وهو أحد الكائنات المهمة لجدا لمعظم التطبيقات ..

أولا : نظرة عامة على الكائن Threads

و هو أحد الكائنات يتم استخدامة فى الجافا ليسمح للتطبيق بالقيام بعدة مهام مترابطة تعمل سويا بشكل متزامن و التى تعمل في الخلفية لتوفير المساحة في ذاكرة الجهاز ....دعونا نأخذ مثالا على ذلك. نفترض أنك تقوم بتطوير برنامج محرر نصوص... في هذه الحالة...يمكنك أن تجعل مشروعك يقوم بعدة مهام فى وقت واحد.....بأن تجعل التطبيق يسمح للمستخدم بالكتابة فى المحرر و أن يقوم التطبيق بالتدقيق الإملائي للنص الذى كتبته فى نفس الوقت قيد التشغيل ...

مثال أخر .... نفترض أنك تقوم بتطوير برنامج تحميل ملفات من الانترنت .....بدون إستخدام هذا الكائن سيقوم التطبيق بتحميل الملف لكنه لن يستطيع تحميل ملفات أخرى الى أن ينتهى من تحميل الملف الحالى ....أما أذا أستخدمت الكائن Threads تستطيع تحميل أكثر من ملف فى وقت واحد ...

ثانيا : كيفية أستخدام الكائن Threads

فيمكنك أستخدام هذا الكائن من خلال توسيع الفئة المراد أستخدامه فيها Threads هناك خياران لديك لأستخدام الكائن عن طريق الامر extends Thread أو من خلال تنفيذ واجهة "Runnable" عن طريق الامر implements Runnable

1_ أستخدام الكائن من خلال توسيع الفئة extending Thread و تتطلب تلك الطريقة ما يلى :

أ_ يتم اضافة الامر extends Thread الى الشفرة الخاصة لانشاء فئة معينة

ب_ يجب أنشاء الطريقة void run و بداخلها الاوامر التى تريد تنفيذها بشكل متزامن .

ج_فى الدالة الرئيسة يتم إنشاء كائن من الفئة أعلاه

د_فى الدالة الرئيسية يتم استدعاء الأسلوب "start" من الكائن الذى تم إنشاؤه.

الآن الكائن Threads جاهز للاستخدام وسوف يبدأ تنفيذ الاوامر التى بداخل الطريقة void run بشكل متواز .....

مثال : في هذا المثال، سوف نقوم بإنشاء اثنين من Threads هما "FirstThread" و "SecondThread". وسيقوم كل من هذه الكائنات بطباعة أرقام 1، 2، 3 ... 10 مع تأخير ثانية واحدة في عرض الرقم التالي. سيتم الطبقة ThreadDemo انطلاق هذه المواضيع "FirstThread" و "SecondThread".

الشفرة الخاصة بالكائن الاول :

```
public class FirstThread extends Thread {
public void run(){

for (int i=1; i<=10; i++){
System.out.println( "Messag from First Thread : " +i);

try {
Thread.sleep(1000);
}
catch (InterruptedException  interruptedException)
{
System.out.println(  "First Thread is interrupted when it is  sleeping"
+interruptedException);
}
}
}
}
```

الشفرة الخاصة بالكائن الثانى :

```
public class SecondThread extends Thread
{

public void run()
{
```

```
for (int i=1; i<=10; i++)
{
System.out.println( "Messag from Second Thread : " +i);

try
{
Thread.sleep (1000);
}
catch (InterruptedException interruptedException)
{

System.out.println( "Second Thread is interrupted when it is sleeping"
+interruptedException);
}
}
}
}
```

الشفرة الخاصة فى الدالة الرئسية :

```
public static void main(String args[])
{

FirstThread   firstThread = new FirstThread();
SecondThread   secondThread = new SecondThread();
firstThread.start();
secondThread.start();
}
}
```

2_ أستخدام الكائن من خلال تنفيذ واجهة "Runnable" و تتطلب تلك الطريقة ما يلى :

أ_يتمَ إضافة الامر implements Runnable الى الشفرة الخاصة لبناء الفئة

ب_ يجب أنشاء الطريقة void run و بداخلها الاوامر التى تريد تنفيذها بشكل متزامن

ج_فى الدالة الرئيسة يتم إنشاء كائن من الفئة أعلاه

د_تخصيص كائن thread للكائن الموجود لدينا

هـ_فى الدالة الرئيسية يتم استدعاء الأسلوب "start" من الكائن الذى تم إنشاؤه

مثال : نفس المثال السابق سنقوم بعملة بتلك الطريقة :

الكائن الاول :

```
public class FirstThread implements Runnable {
public void run(){
for (int i=1; i<=10; i++){
System.out.println( "Messag from First Thread : " +i);
try {
Thread.sleep(1000);
}
catch (InterruptedException  interruptedException)
{
System.out.println(  "First Thread is interrupted when it is  sleeping"
+interruptedException);
}
}
}
}
```

الشفرة الخاصة بالكائن الثانى :

```
public class SecondThread extends Thread
```

```
{
public void run()
{
for (int i=1; i<=10; i++)
{
System.out.println( "Messag from Second Thread : " +i);
try
{
Thread.sleep (1000);
}
catch (InterruptedException interruptedException)
{
System.out.println( "Second Thread is interrupted when it is sleeping"
+interruptedException);
}
}
}
}
```

الدالة الرئسية :

```
public static void main(String args[])
{

FirstThread   firstThread = new FirstThread();

SecondThread   secondThread = new SecondThread();

Thread thread1 = new Thread(firstThread);
thread1.start();
```

```java
Thread thread2 = new Thread(secondThread);
thread2.start();
}
}
```

لاحظ الفرق بين الطريقتين فى الدالة الرئسية

للمزيد من الدروس :

مشروع إنشاء ساعة رقمية بالجافا ...للتعرف على كيفية استخدام الكائن Timer

طريقة إنشاء متصفح إنترنت بسط للغاية بالجافا

تعرف على طرق و أسالب الوقت و التاريخ فى لغة الجافا و كيفية تنسيقهما

خطوة بخطوة لبرمجة مدير تحميل المفات EgyDownloadManager بلغة الجافا واستخدام NetBeans

تعلم كيفية إرسال و إستقبال البريد الالكترونى مع المرفقات و تنسق html باستخدام الجافا

تعلم كيفية إنشاء تطبق دردشة بسط للجوالات بأستخدام J2ME مع NetBeans

كل ذلك و أكثر فى منتدى مصر التقنى

http://egy-tech.forumegypt.net